

# System Requirements RU Staying

Software Engineering

01:332:452

Report 1: Part 2

## **By Group #11**

Keya Patel

Zain Sayed

Mohammed Sapin

Purna Haque

Nga Man (Mandy) Cheng

Rameen Masood

Shilp Shah

Mathew Varghese

Thomas Tran

Eric Zhang

Github: <https://github.com/mohammedsapin/RUStaying>

Submission Date: Feb 17, 2019

### Section 3: Functional Requirements Specification

#### Stakeholders

The main stakeholders of this app are hotel owners and managers. The app is commissioned by the owners of the hotel in an attempt to expand the hotel brand and earn more profit. With our design, this app can significantly improve the efficiency of the hotel services, which in turn increases the number of guests, therefore increasing profit. As investors, the hotel owners are certainly interested in the growth and success of the app. The hotel managers are also stakeholders because they control the day-to-day activities of the hotel and it is up to them to use the data our app provides to efficiently schedule staff and hotel services. Also, all the users and guests of the app are stakeholders too because through their support and feedback, we can continue to improve the features and user interface. And finally, all the developers of the RUStaying app since we are investing a lot of time and effort into the creation of the system. As developers, our role is to ensure the success of the app and continue to make improvements based on the feedback from the other stakeholders.

#### Actors and Goals

##### Key

I → Initiating

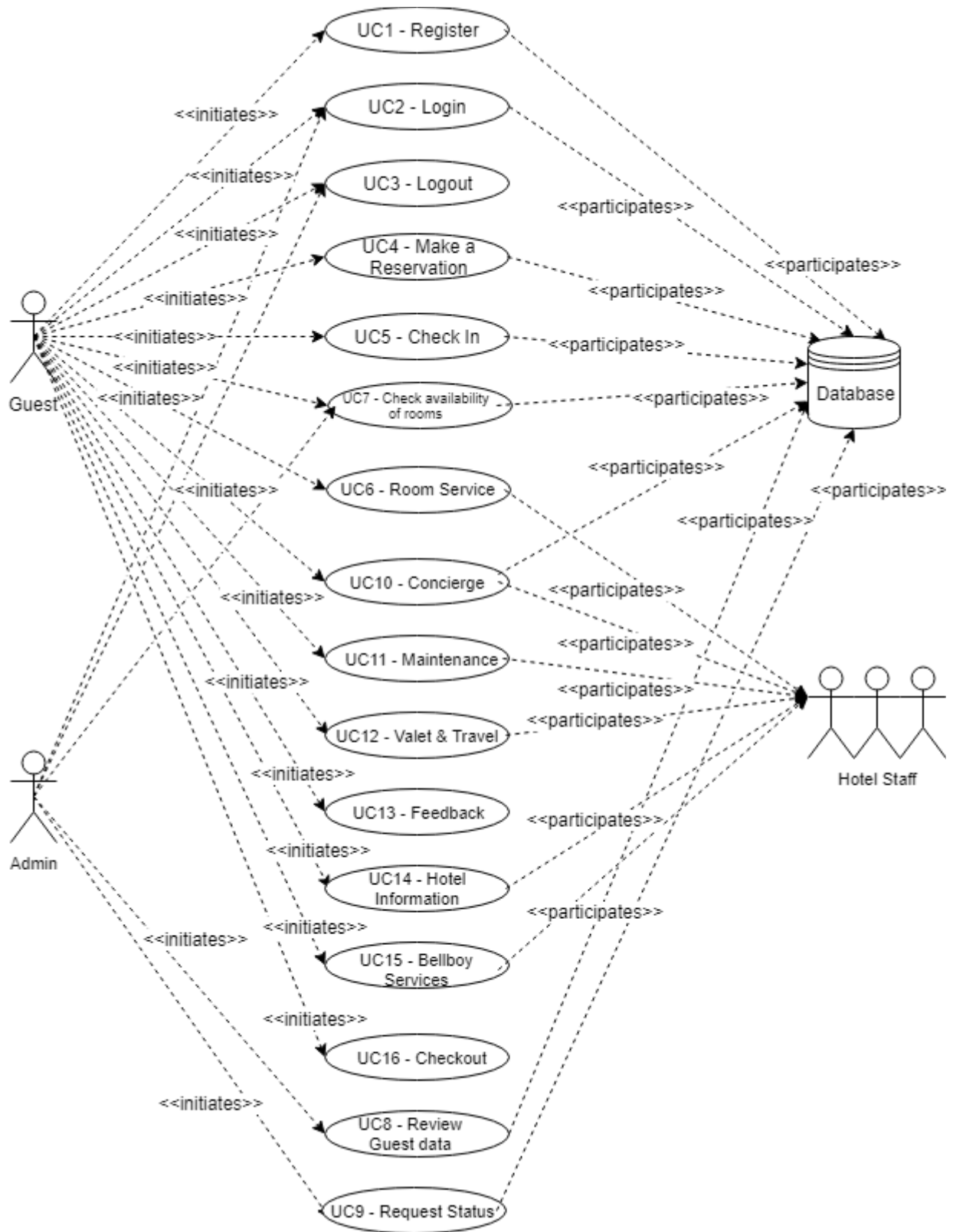
P → Participating

Actors	Goals
Guests - I	Will be able to login and send requests to Hotel Staff as needed
Admins/Manager- I	Primary initiating goal is to be able to <u>monitor</u> customer interaction with hotel services. Will be able to log in and view requests as they need.
Hotel Staff (includes Manager) - P	Guests will be sending multiple requests for various services throughout the day. Staff receives and will act accordingly upon request. Restaurant will view orders, maintenance can view forms,etc..
Database - P	Will store all essential information from guests and general hotel information. This will be an efficient way to organize and access necessary info on guests.

**Use Cases:**

The RUStaying app has 2 main users, the guest and admin. Each user has their own separate use cases

1. Register (guest) - to register an account on the application
2. Login (Guest & admin) - to log into a user account
3. Logout (Guest & admin) - to log out of a user account
4. Make a reservation (Guest) - to make a reservation for a room
5. Check In (Guest) - to check into the hotel
6. Room Service (Guest) - to call for room service
7. Check availability of Rooms (Admin & Guest) - to check which rooms are in use or vacant
8. Review Guest Data (Admin) - to check the information of guests, given a specific room
9. Request Status (Admin) - to be able to change or mark a guest request “in progress” or “granted”
10. Concierge (Guest) - access front desk information through the app
11. Maintenance (Guest) - to call for maintenance
12. Valet & Travel (Guest) - to call for a car to the airport or wherever the guest wants to go, or call for a valet to park the guest’s car
13. Feedback (Guest) - to allow the guest to provide feedback after their stay
14. Hotel Information (Guest) - to give guests a visual of the restaurant menu, gym hours, pool hours, and spa hours
15. Request Bellboy Services (Guest) - to call a bellboy to get guest’s luggage
16. Checkout (Guest) - to check out of the hotel



## Use Case Descriptions

<b>Use Case 1: Register an account</b>
Related Requirements: REQ-1
Initiating Actors: Application User
Actor's Goal: To create an account on RUStaying app
Participating Actors: Database (keeping track of past emails used)
Preconditions: <ol style="list-style-type: none"><li>1. The email the Guest is using to register has not been used previously in our system</li><li>2. The email and password meet certain criteria for security</li></ol>
Postconditions: User has an account to access the features of the app
Flow of events for main success scenarios: <ol style="list-style-type: none"><li>1. The guest downloads the app</li><li>2. Choose "Register an Account"</li><li>3. Fills out the required account information</li><li>4. Check for valid email and password</li><li>5. Store guest data in database</li><li>6. New account is successfully created</li></ol>

<b>Use Case 2: Login to an account</b>
Related Requirements: REQ-1, REQ-2, REQ-22
Initiating Actors: Application User
Actor's Goal: To log into an existing account created on RUStaying App
Participating Actors: Database
Preconditions: <ol style="list-style-type: none"><li>1. User has an already existing account</li></ol>
Postconditions: User is able to back into his/her account
Flow of events for main success scenarios: <ol style="list-style-type: none"><li>1. The user opens the app</li><li>2. The user fills in the email and password</li><li>3. If email/password are incorrect, system notifies user</li><li>4. The user successfully logs into the desired account</li></ol>

<b>Use Case 3: Logout</b>
Related Requirements: REQ-1, REQ-2, REQ-3
Initiating Actors: Application User
Actor's Goal: To exit their account in the app
Participating Actors: Database
Preconditions: <ul style="list-style-type: none"> <li>1. The user already has an existing account</li> <li>2. The user is currently logged into the system</li> </ul>
Postconditions:
Flow of events for main success scenarios: <ul style="list-style-type: none"> <li>1. The user initiates logout function</li> <li>2. Database confirms user account</li> <li>3. Database marks user account as logged out</li> </ul>

<b>Use Case 4: Make a reservation</b>
Related Requirements: REQ-1, REQ-2, REQ-4, REQ-6, REQ-8, REQ-23
Initiating Actors: Application User/Guest
Actor's Goal: To book a room for the duration of their choosing and/or luxury of their choice
Participating Actors: Database (to check for accommodations)
Preconditions: <ul style="list-style-type: none"> <li>1. Guest must have created an account with our app</li> <li>2. Guest has successfully logged in</li> </ul>
Postconditions: Guest will receive confirmation via app
Flow of events for main success scenarios: <ul style="list-style-type: none"> <li>1. User has opened app on device and has logged in</li> <li>2. User selects the option "Make a Reservation"</li> <li>3. User fills out details of what type of room they are looking for (price, number of people, size of room, etc..)</li> <li>4. System will relay information and gather available rooms and present options to user</li> <li>5. User will choose an option</li> <li>6. System has successfully booked the room for user</li> </ul>

<b>Use Case 5: Check In</b>
Related Requirements: REQ-3, REQ-10, REQ-23
Initiating Actors: User
Actor's Goal: User would like to check in for their room upon arrival
Participating Actors: Hotel Staff, database (to check for reservation)
Preconditions: User has created an account and made a reservation
Postconditions: User received digital room key and can access services.
<p>Flow of events for main success scenarios:</p> <ol style="list-style-type: none"> <li>1. user creates an account</li> <li>2. System adds user to database</li> <li>3. user logs in to app</li> <li>4. System verifies user through database</li> <li>5. click make a reservation</li> <li>6. System pulls from database availability of rooms</li> <li>7. User reserves a room at the hotel</li> <li>8. System updates database to include User's reservation</li> <li>9. User checks in on app upon arrival</li> <li>10. System verifies user and provides user with a digital room key.</li> </ol>

<b>Use Case 6: Room Service</b>
Related Requirements:REQ-2, REQ-3, REQ-5
Initiating Actors: Application User/ Guest
Actor's Goal: To be able to call room service for a variety of purposes (Clean the room, replace toiletries, replace bed sheets, etc. )
Participating Actors: Room Staff, Maids
Preconditions: 1. Guest has been checked-in before requesting any service
Postconditions: Guest will be notified when request has been sent
Flow of events for main success scenarios: 1. User has logged in and has selected "Request Room Service" 2. System will confirm check in status of guest and return with a list of services available to guest 3. User will choose an option along with the timing of when it is needed 4. System will confirm and send notification to team responsible for service

<b>Use Case 7: Check Availability of Rooms</b>
Related Requirements: REQ - 6, REQ - 13
Initiating Actors: Guest, Admin
Actor's Goal: User would like to check the availability of the rooms
Participating Actors: Guest Admin
Preconditions: Users have an account created
Postconditions: App displays availability of rooms
Flow of events for main success scenarios: 1. User logs in to app 2. System verifies user has an account 3. User clicks make a reservation 4. System pulls from database availability of rooms 5. User is able to see which rooms are vacant.



<b>Use Case 8: Review Guest / Service Data</b>
Related Requirements: REQ-13
Initiating Actors: Admin
Actor's Goal: To view the guest data, room availability and usage of hotel services
Participating Actors: Database
Preconditions: <ol style="list-style-type: none"> <li>1. Accounts created by guests</li> <li>2. Guests request hotel services from app</li> </ol>
Postconditions: <ol style="list-style-type: none"> <li>1. Display guests data (ex: Number of guests in hotel)</li> <li>2. A breakdown of how much each service is used</li> <li>3. List of rooms available / occupied</li> </ol>
Flow of events for main success scenarios: *This use case is for all data storage so it <<includes>> all other use cases* <ol style="list-style-type: none"> <li>1. Guests continue to use services through the app</li> <li>2. Data is kept track of and it sent to the database to store</li> <li>3. The data is collected from the database frequently</li> <li>4. The data is presented to the Admin in a user friendly format</li> </ol>

<b>Use Case 9: Request Status</b>
Related Requirements: REQ-5, REQ-7, REQ-9, REQ-12, REQ-13
Initiating Actors: Admin
Actor's Goal: To be able to change or mark a guest request "in progress" or "granted"
Participating Actors: Guest, Hotel Staff, Database of Hotel Staff Activities
Preconditions: <ol style="list-style-type: none"> <li>1. Guest must have made a request for some type of service, see REQ-5, REQ-7, REQ-9, and REQ-12</li> </ol>
Postconditions: <ol style="list-style-type: none"> <li>1. Guest request will be either labeled "in progress" or "granted"</li> </ol>
Flow of events for main success scenarios: <ol style="list-style-type: none"> <li>1. User has opened the app and logged in</li> <li>2. User makes a request (any that are offered in the app)</li> <li>3. System receives request and labels the request as "in progress"</li> <li>4. User reports to the system when service has been completed</li> <li>5. System marks requests as "granted"</li> </ol>

<b>Use Case 10: Concierge</b>
Related Requirements: REQ - 17
Initiating Actors: User
Actor's Goal: To use the concierge service to obtain hotel information or other general information
Participating Actors: System
Preconditions: <ol style="list-style-type: none"> <li>1. User will type a question into the concierge tab, OR</li> <li>2. Select a question from general topics listed in the tab</li> </ol>
Postconditions: <ol style="list-style-type: none"> <li>1. User will be presented with the information they requested, OR</li> <li>2. Be given the phone number of an in-person concierge at the hotel's front desk</li> </ol>
Flow of events for main success scenarios: <ol style="list-style-type: none"> <li>1. Guest has opened the app and logged in</li> <li>2. Guest selects the concierge tab</li> <li>3. Guest enters a question or selects one from the general questions provided</li> <li>4. System receives the information and processes it and displays an answer <ol style="list-style-type: none"> <li>a. If the system does not have an answer to the guest's question, the phone number of the concierge at the hotel's front desk will be displayed</li> </ol> </li> </ol>

<b>Use Case 11: Maintenance</b>
Related Requirements: REQ-2, REQ-7
Initiating Actors: Guest
Actor's Goal: To submit a maintenance request into the app and for a hotel worker to solve the issue
Participating Actors: System, Hotel Staff, Maintenance Worker
Preconditions: <ol style="list-style-type: none"> <li>1. Guest is logged in</li> <li>2. Guest needs to submit maintenance request</li> </ol>
Postconditions: <ol style="list-style-type: none"> <li>1. Guest will have maintenance requests fulfilled</li> <li>2. Confirmation message will pop up after fulfilled</li> </ol>
Flow of events for main success scenarios: <ol style="list-style-type: none"> <li>1. User has logged in</li> <li>2. Database verifies user has account</li> <li>3. App directs user to dashboard</li> <li>4. User selects Maintenance Requests</li> <li>5. Database pulls options for Maintenance Requests</li> <li>6. User will be able to select from the options presented</li> <li>7. App will submit option into database</li> <li>8. Database will send confirmation message that request was successfully submitted</li> </ol>

<b>Use Case 12: Valet &amp; Travel</b>
Related Requirements: REQ 12, REQ 25
Initiating Actors: Guest
Actor's Goal: To call for a car to the airport or wherever the guest wants to go, or call for a valet to park the guest's car
Participating Actors: Hotel Staff, Manager
Preconditions: <ol style="list-style-type: none"> <li>1. Guest is logged in</li> <li>2. Guest has checked in</li> </ol>
Postconditions: <ol style="list-style-type: none"> <li>1. Guest will have their requested car waiting outside</li> <li>2. Guest will be notified once the car is called and ready</li> </ol>
Flow of events for main success scenarios: <ol style="list-style-type: none"> <li>1. User logs on</li> <li>2. User checks in</li> <li>3. User clicks guest services</li> <li>4. User can select car service and choose to call a car</li> <li>5. Car Services will indicate that they have arrived for pick up</li> <li>6. System will notify User that car has arrived and is ready</li> </ol>

<b>Use Case 13: Feedback</b>
Related Requirements: REQ-14
Initiating Actors: Guest
Actor's Goal: To allow the user to input feedback upon check out
Participating Actors: System, Database
Preconditions: <ul style="list-style-type: none"> <li>1. User has checked out of the hotel</li> </ul>
Postconditions: <ul style="list-style-type: none"> <li>1. Check out confirmation will pop up</li> <li>2. Feedback form will pop up</li> </ul>
Flow of events for main success scenarios: <ul style="list-style-type: none"> <li>1. User has hit check out</li> <li>2. System will output a check out confirmation</li> <li>3. System will output a feedback form from Database</li> <li>4. User will fill out or dismiss feedback form and submit it into the Database</li> <li>5. System will output a submission confirmation</li> </ul>

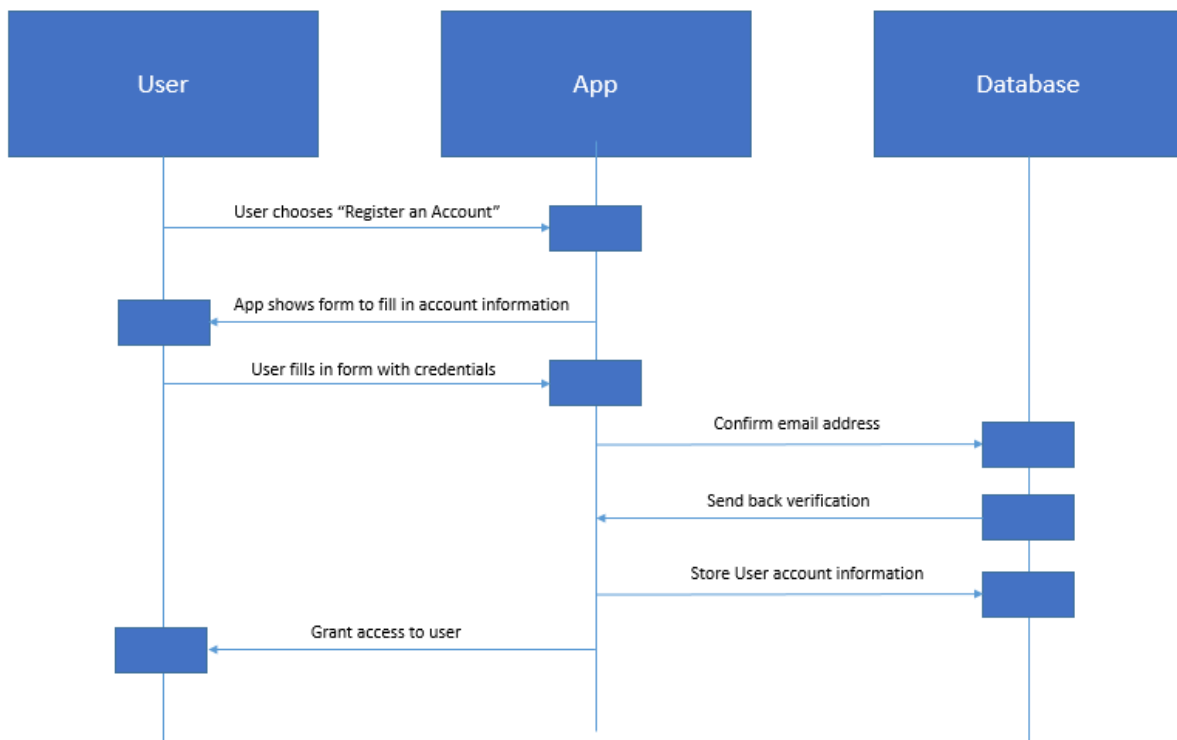
<b>Use Case 14: Hotel Information</b>
Related Requirements: REQ-11
Initiating Actors: Application User
Actor's Goal: To view restaurant menus, gym hours, pool hours, spa hours at the user's convenience, etc...
Participating Actors: Database
Preconditions: <ul style="list-style-type: none"> <li>1. Guest is logged in</li> </ul>
Postconditions: <ul style="list-style-type: none"> <li>1. Guest will be informed of hotel information</li> </ul>
Flow of events for main success scenarios: <ul style="list-style-type: none"> <li>1. Guest has logged in</li> <li>2. Guest clicks to view hotel information</li> <li>3. System will display hotel information</li> </ul>

<b>Use Case 15: Request Bellboy Services</b>
Related Requirements: REQ-9 REQ-25
Initiating Actors: Guest
Actor's Goal: get their luggage moved to their room
Participating Actors: Hotel Staff, Manager
Preconditions: <ol style="list-style-type: none"> <li>1. Guest is logged in</li> <li>2. Guest has checked in</li> </ol>
Postconditions: <ol style="list-style-type: none"> <li>1. Guest will have their luggage taken up to their room</li> <li>2. Guest will be notified once luggage has been taken up</li> </ol>
Flow of events for main success scenarios: <ol style="list-style-type: none"> <li>1. User logs on</li> <li>2. User checks in</li> <li>3. User clicks guest services</li> <li>4. User can select bellboy and choose to move luggage to room</li> <li>5. Bellboy will indicate that they have finished moving luggage</li> <li>6. System will notify User that luggage has been taken up to their room</li> </ol>

<b>Use Case 16: Checkout</b>
Related Requirements: REQ-16
Initiating Actors: Application User
Actor's Goal: To check out of the hotel through the app
Participating Actors: Hotel Staff, Database
Preconditions: <ul style="list-style-type: none"> <li>1. Guest has finished their stay at the hotel</li> </ul>
Postconditions: <ul style="list-style-type: none"> <li>1. Guest will be able to write a feedback report</li> </ul>
Flow of events for main success scenarios: <ul style="list-style-type: none"> <li>1. User finishes their stay at the hotel</li> <li>2. User checks out of hotel through the app</li> <li>3. User is given the option to have a bellboy help move luggages to the main lobby</li> <li>4. System is notified that room service is needed</li> <li>5. System accesses database to update that the room is available</li> </ul>

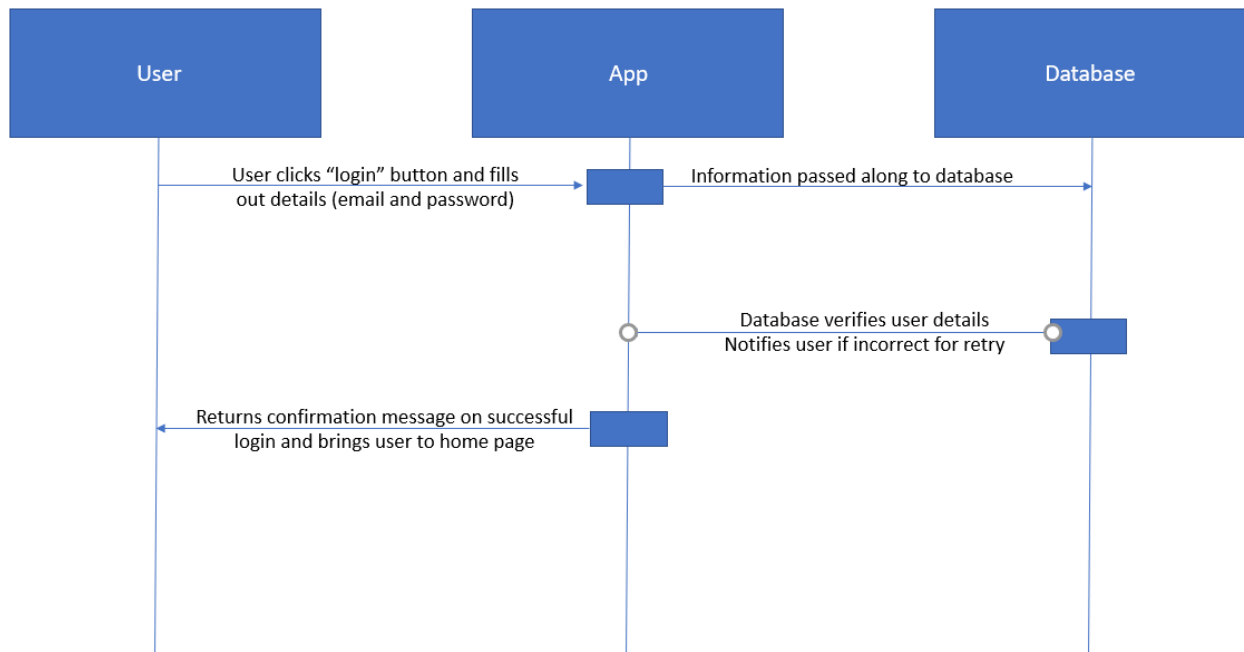
#### Use Case Diagram

#### **Use Case 1: Register an Account**

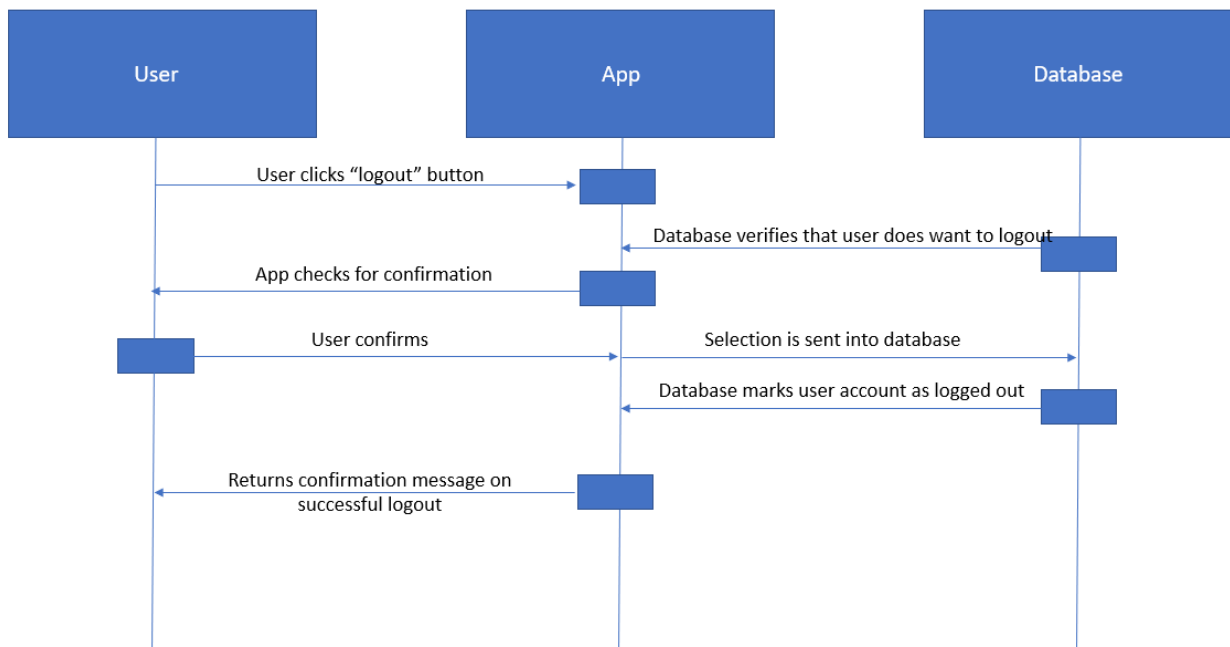




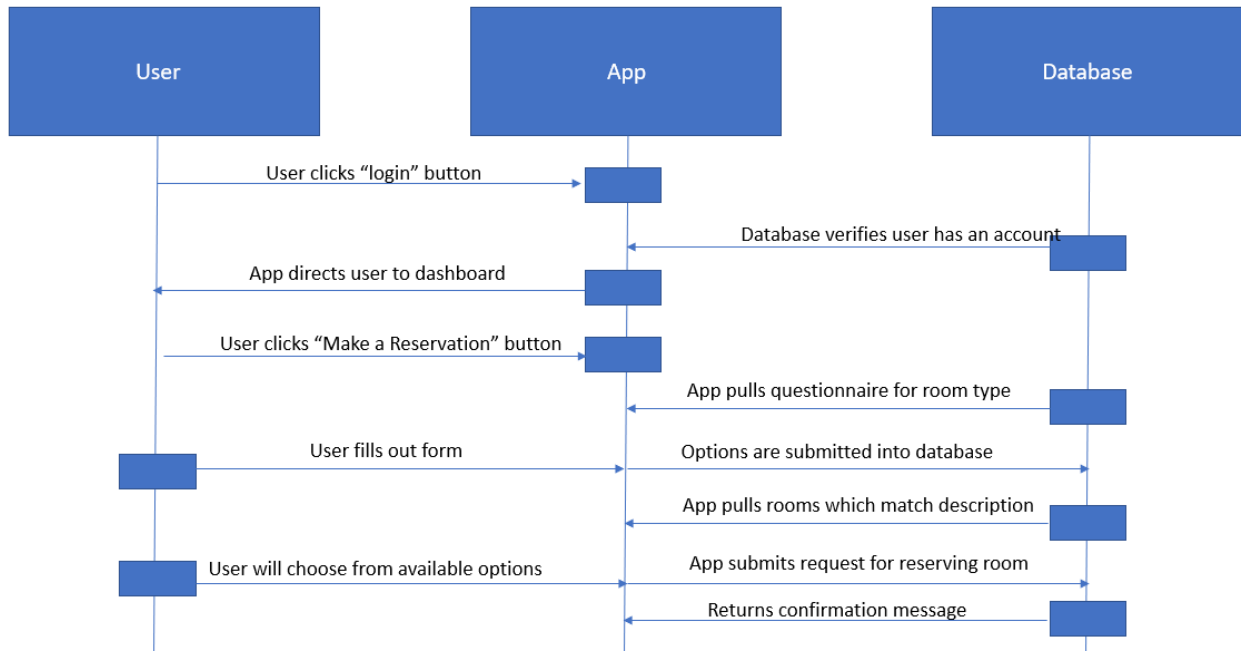
## Use Case 2: Login



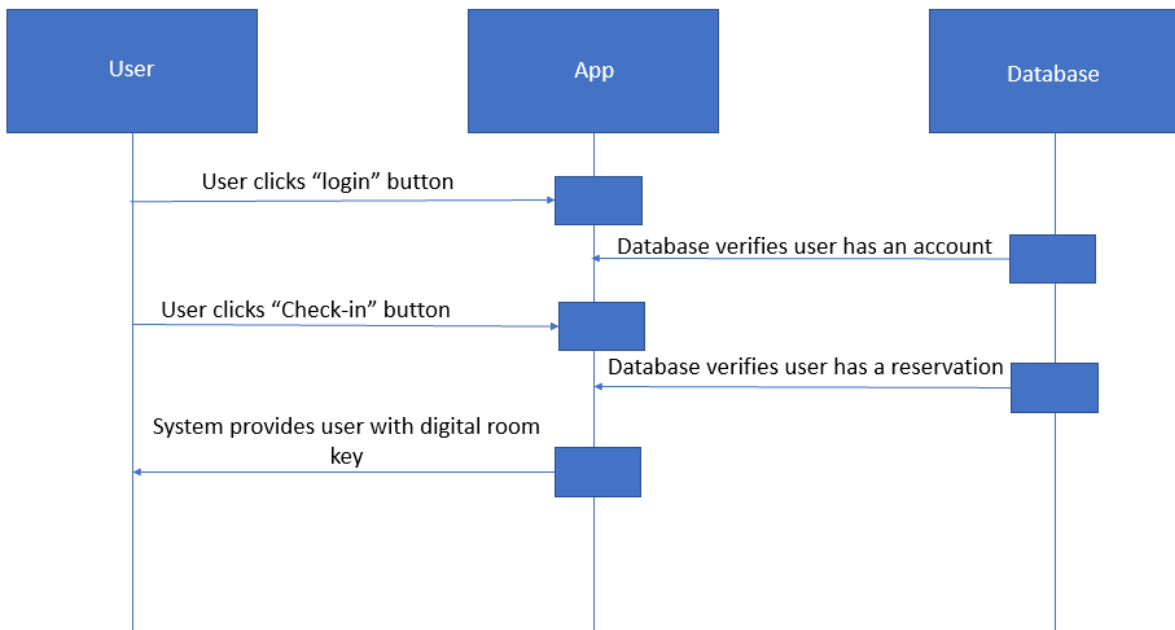
## Use Case 3: Logout



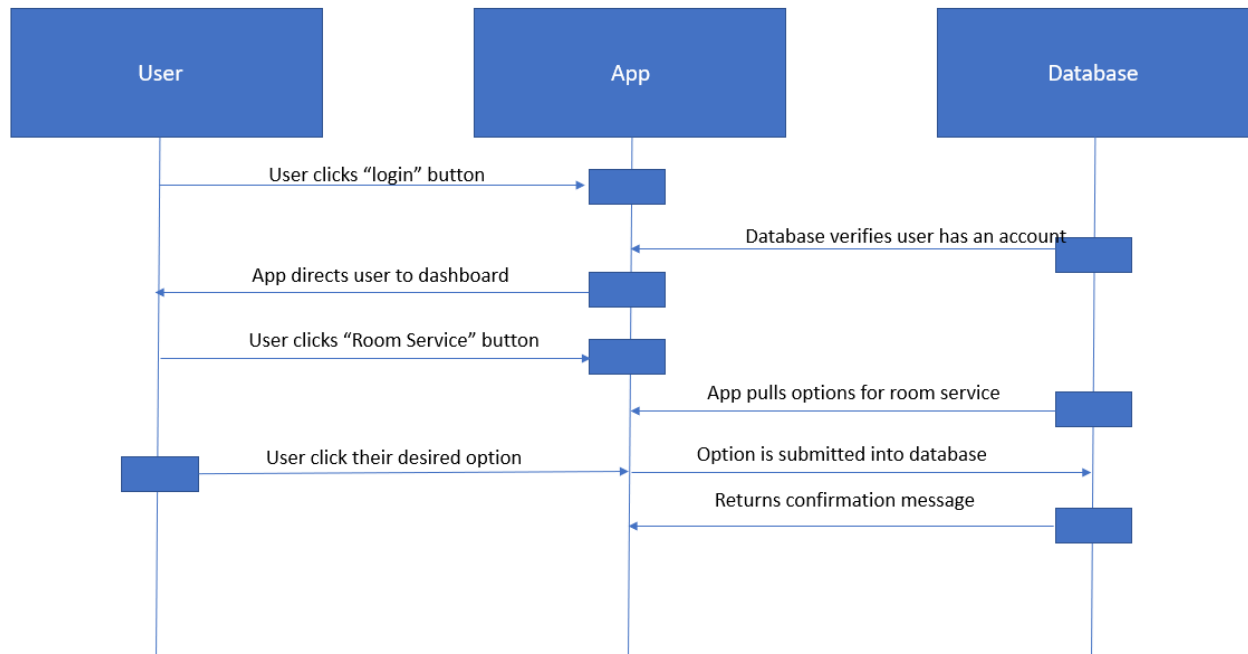
#### Use Case 4: Make a Reservation



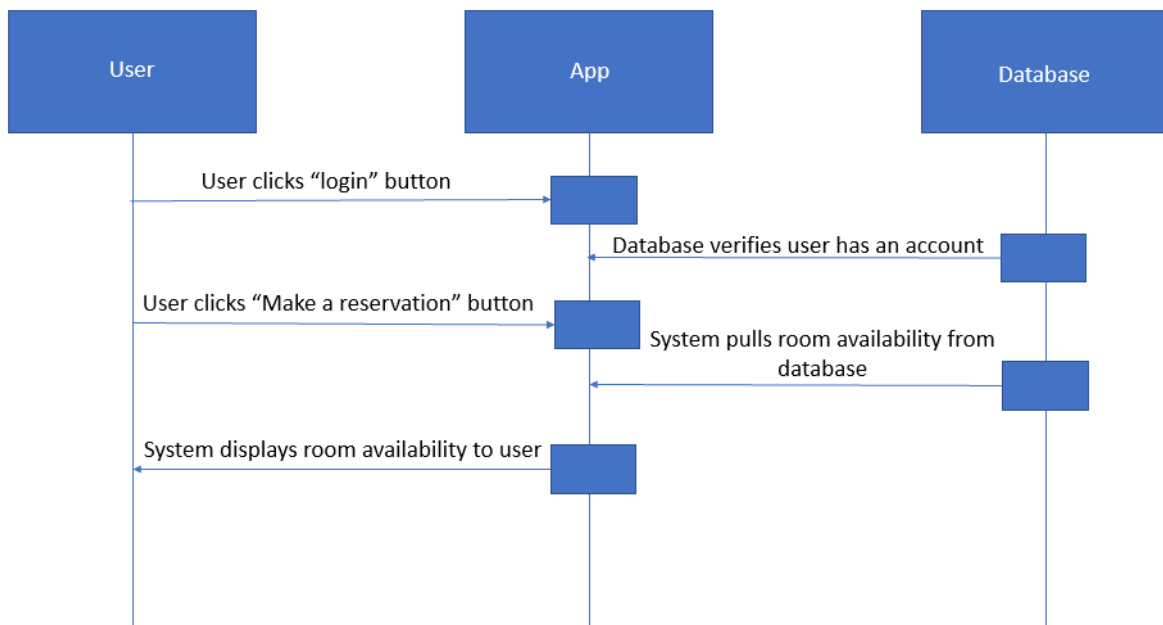
#### Use Case 5: Check-In



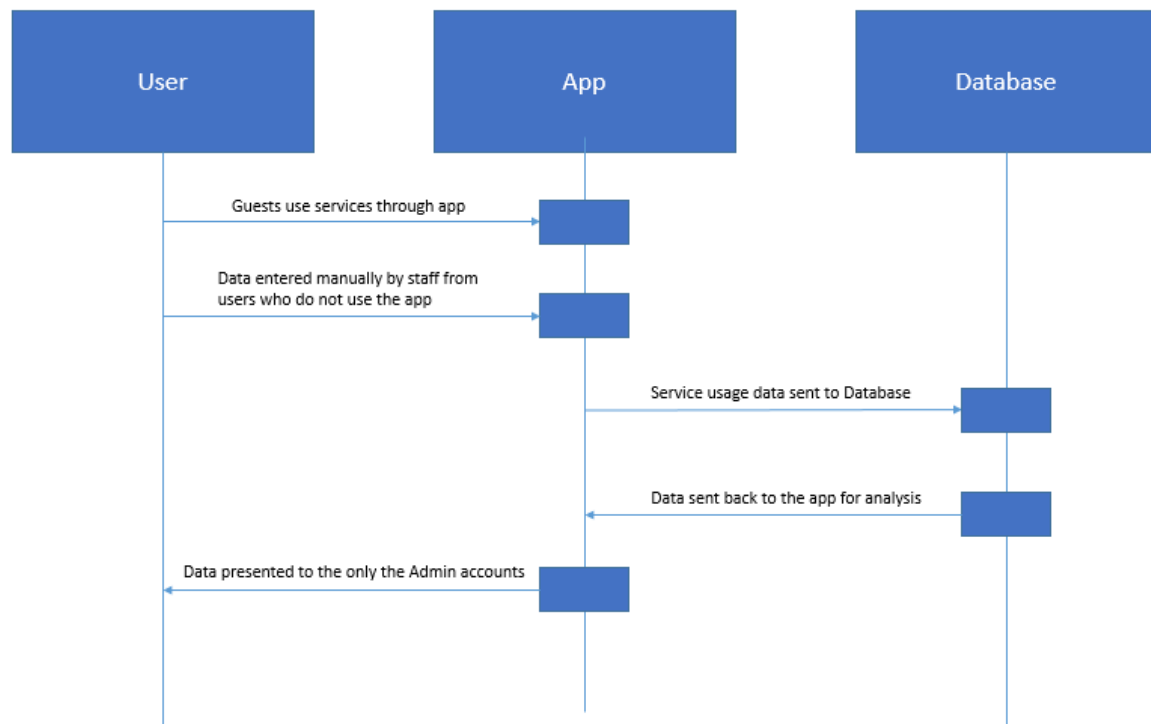
### Use Case 6: Room Service



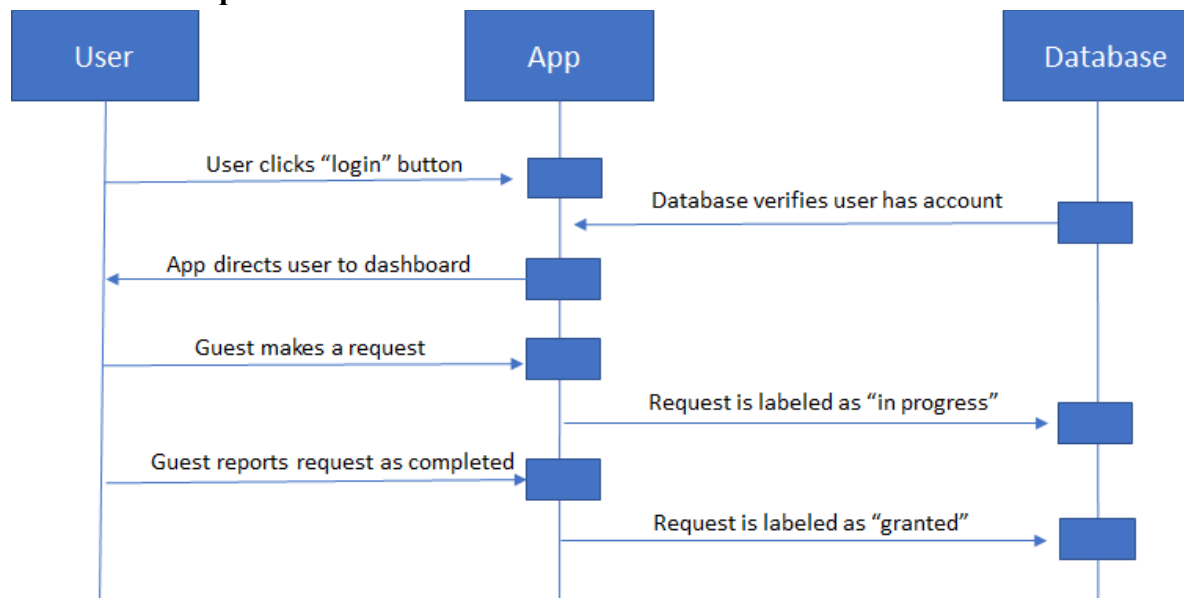
### Use Case 7: Check Availability of Rooms



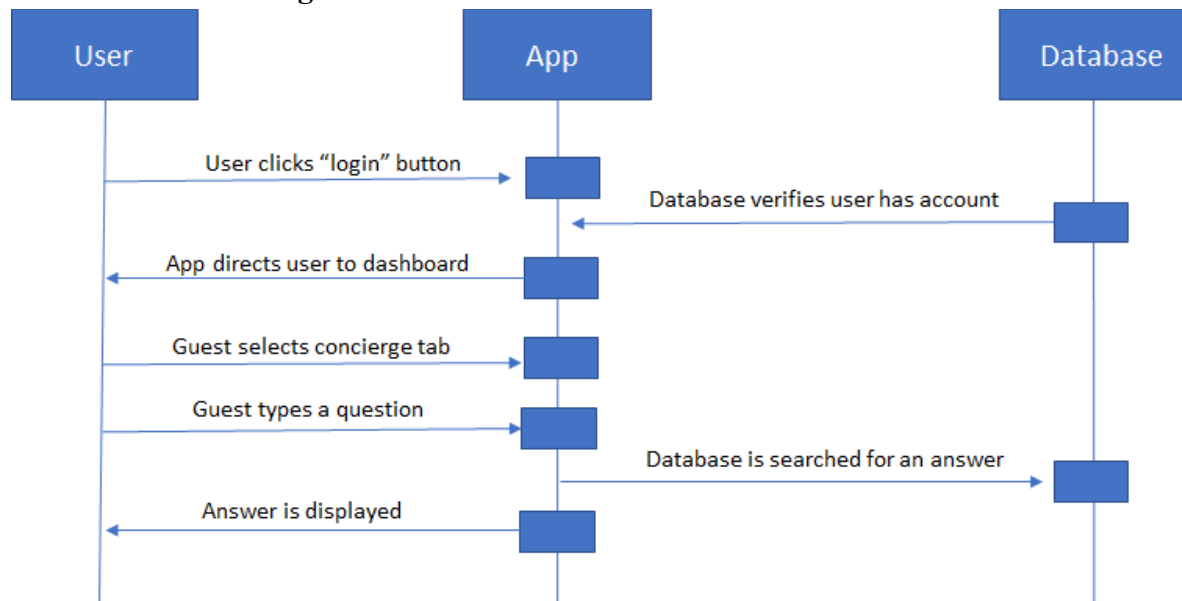
### Use Case 8: Review Guest / Service Data



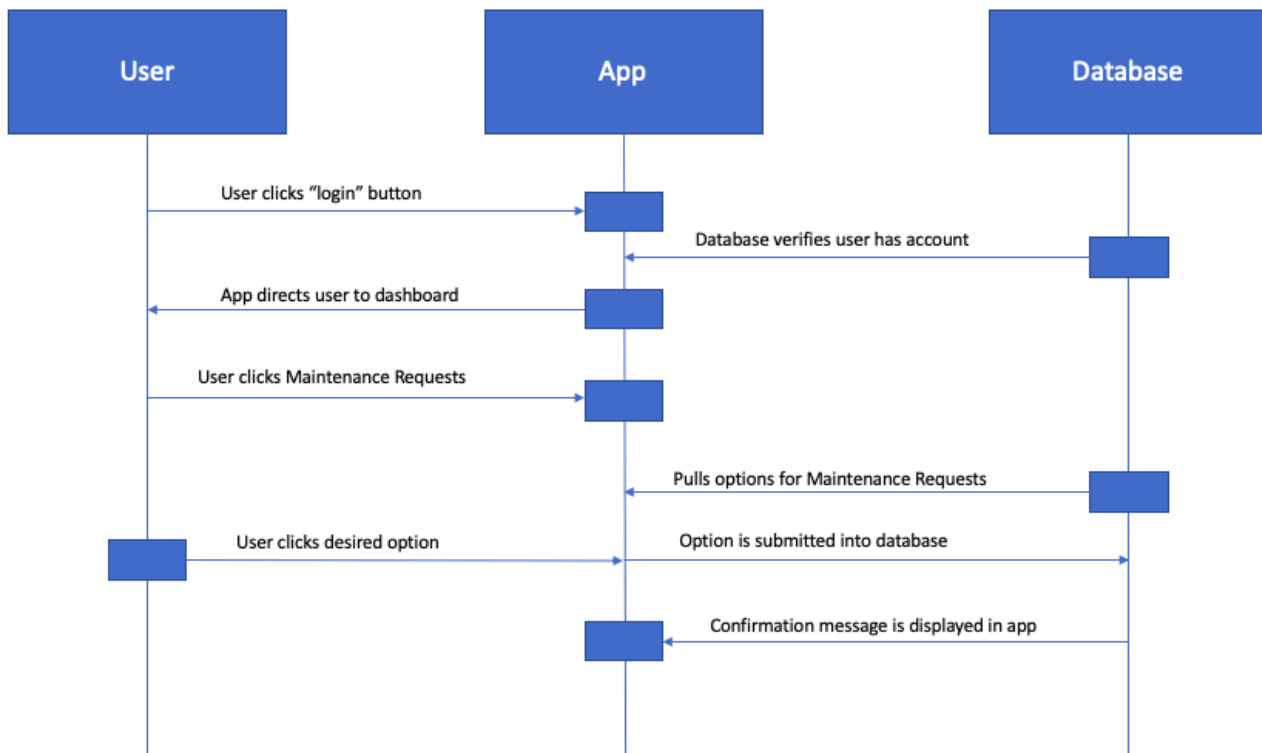
### Use Case 9: Request Status



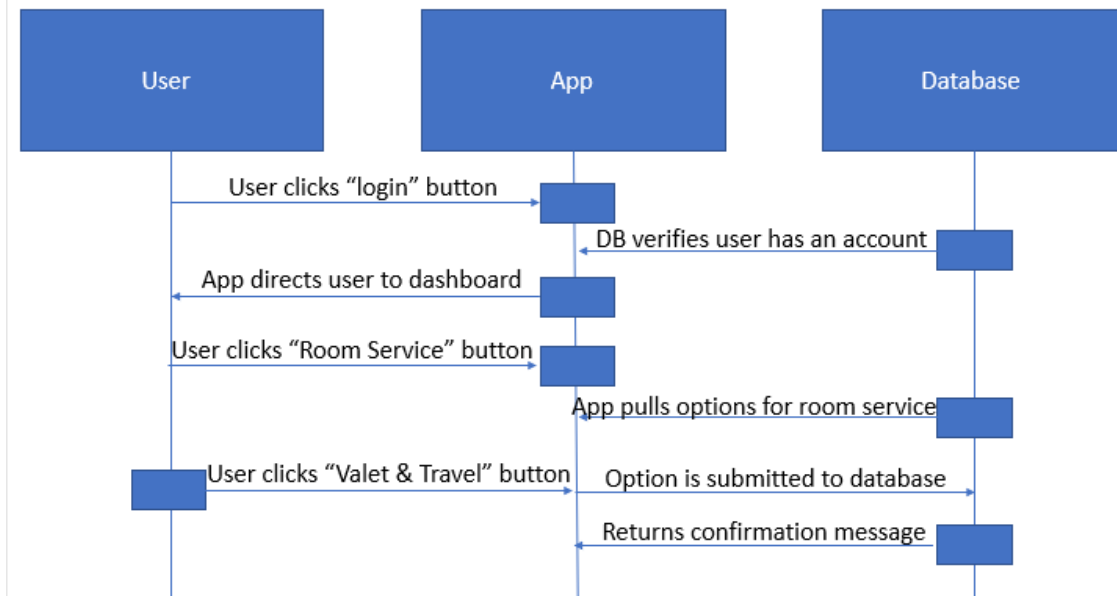
### Use Case 10: Concierge



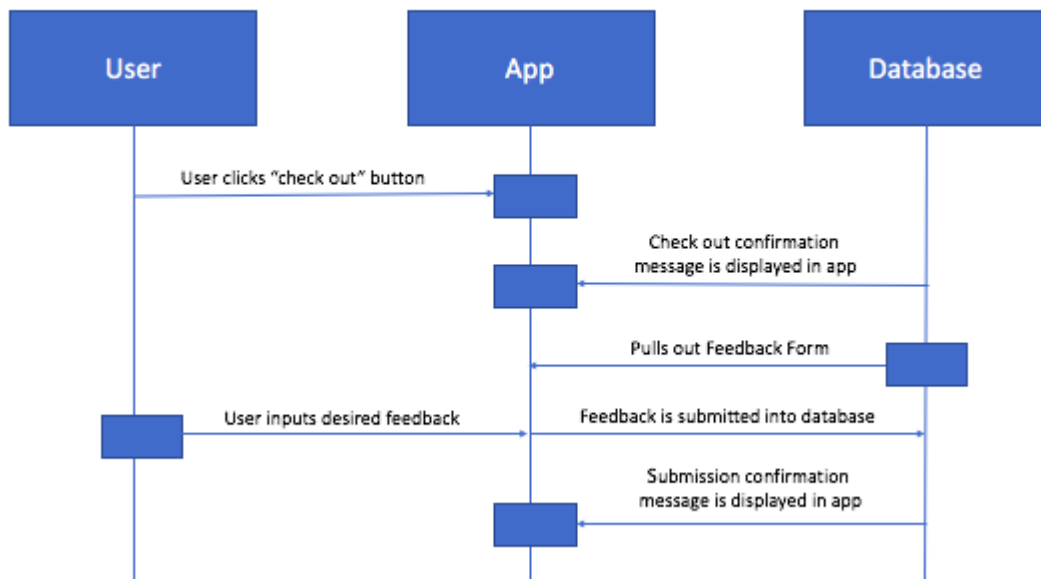
### Use Case 11: Maintenance



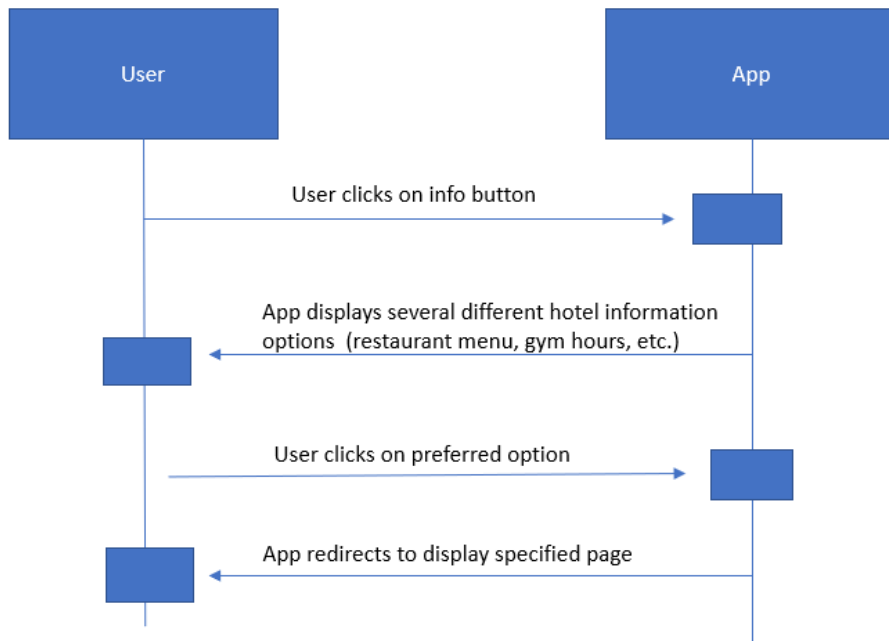
### User Case 12: Valet & Travel



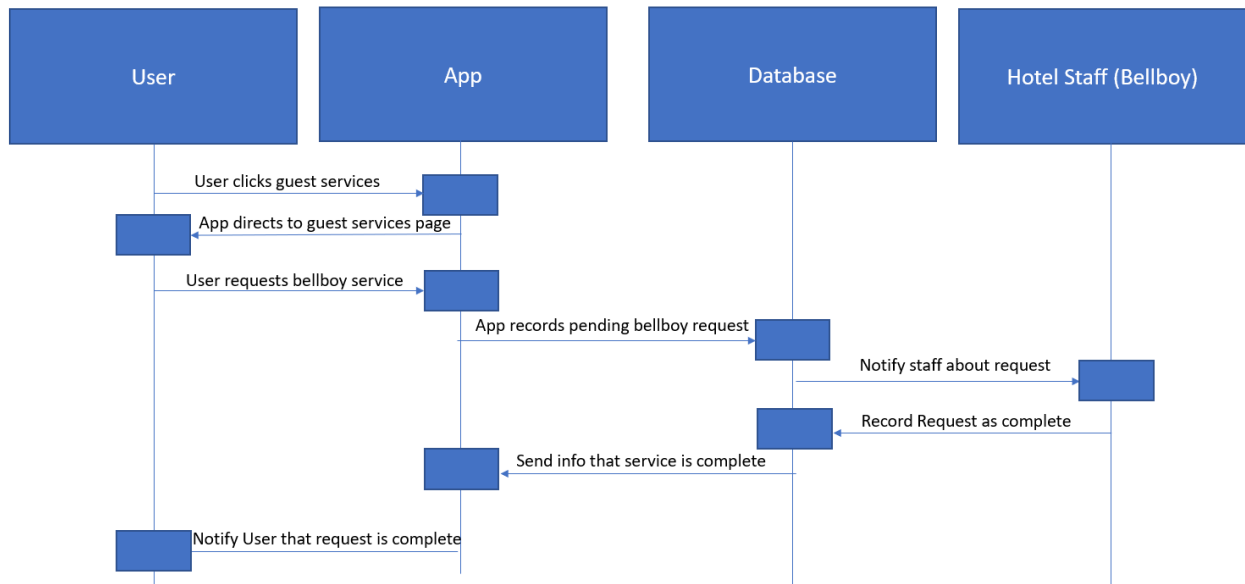
### User Case 13: Feedback



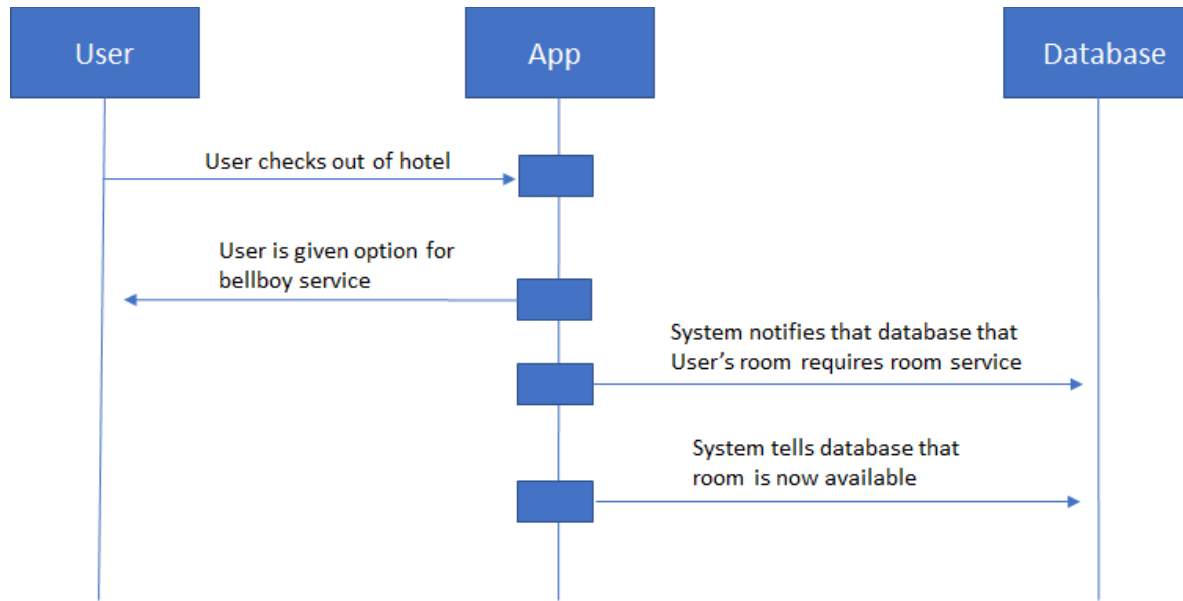
## User Case 14: Hotel Information



## User Case 15: Bellboy Services



## User Case 16: Checkout





Traceability Matrix

Reqs	P.W	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10	UC11	UC12	UC13	UC14	UC15	UC16
1	2	X	X	X	X												
2	2		X	X	X		X					X					
3	6			X		X	X										
4	6				X												
5	4						X			X							
6	8				X			X									
7	5									X		X					
8	3				X												
9	2									X						X	
10	7					X											
11	1														X		
12	3									X			X				
13	10							X	X	X							
14	1													X			
15	3																
16	6																X
17	1										X						
Max P.W.		2	2	6	8	7	6	10	10	10	1	5	3	1	1	2	6
Total P.W.		2	4	10	21	13	12	18	10	24	1	7	3	1	1	2	6

## Section 4

### Use Case 4: Make Reservation

RUSTAYING

Login

Username

Password

Or create a new account

Start page where user can login

Welcome Back, (Name)

Book a room

Check in

Inbox

My Key Card

User selects Book a Room

Make Reservation

Duration Of Stay

Start:

Date: 

February 2019

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28		

End:

Date: 

February 2019

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28		

User selects start and end date

Choose Room Type

List of Rooms Available for  
Dates: 2/28/2019 to 3/4/2019

Single Bedroom

Double Bedroom

Triple

Quad

Queen

King

User selects from available rooms

Confirm Reservation

Confirm your reservation  
for {Start Date} to {End  
Date} in a Single Bedroom

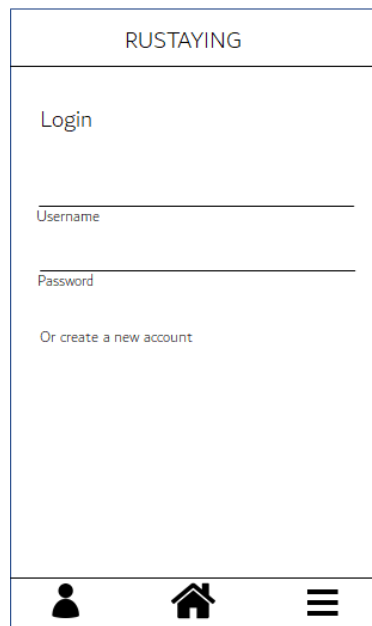
CONFIRM!

Confirmation Number:  
{Number generated}

Text Message  
Confirmation will be sent

User confirms booking

## Use Case 11: Maintenance Request



RUSTAYING

Login

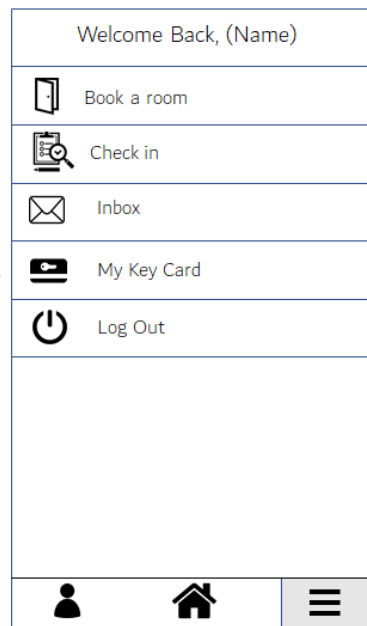
Username

Password

Or create a new account

Bottom navigation bar: User icon, Home icon, Menu icon

User logs in

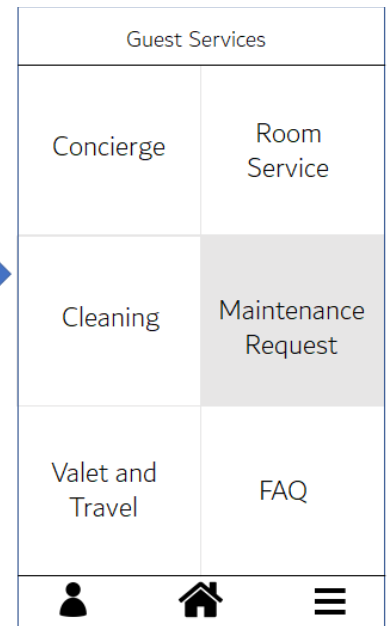


Welcome Back, (Name)

- Book a room
- Check in
- Inbox
- My Key Card
- Log Out

Bottom navigation bar: User icon, Home icon, Menu icon (highlighted)

Select the Guest Services

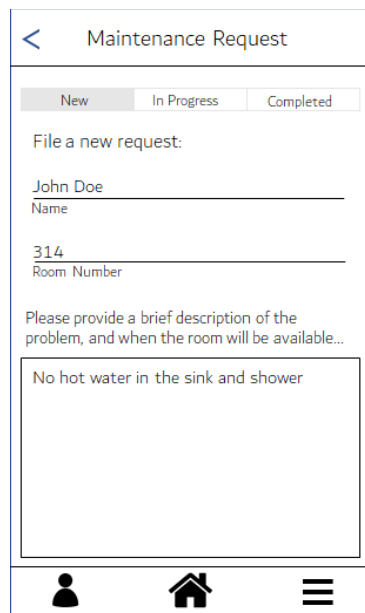


Guest Services

Concierge	Room Service
Cleaning	Maintenance Request
Valet and Travel	FAQ

Bottom navigation bar: User icon, Home icon, Menu icon (highlighted)

Select Maintenance Request



< Maintenance Request

New In Progress Completed

File a new request:

John Doe  
Name

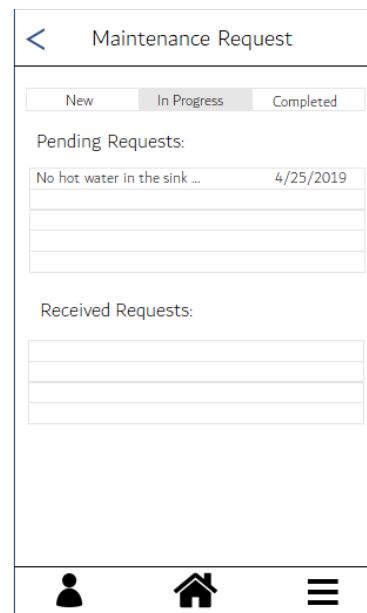
314  
Room Number

Please provide a brief description of the problem, and when the room will be available...

No hot water in the sink and shower

Bottom navigation bar: User icon, Home icon, Menu icon

New Requests



< Maintenance Request

New In Progress Completed

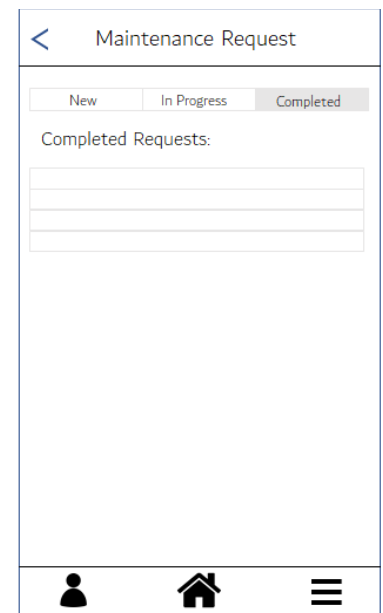
Pending Requests:

No hot water in the sink ... 4/25/2019

Received Requests:

Bottom navigation bar: User icon, Home icon, Menu icon

Pending/In progress Requests



< Maintenance Request

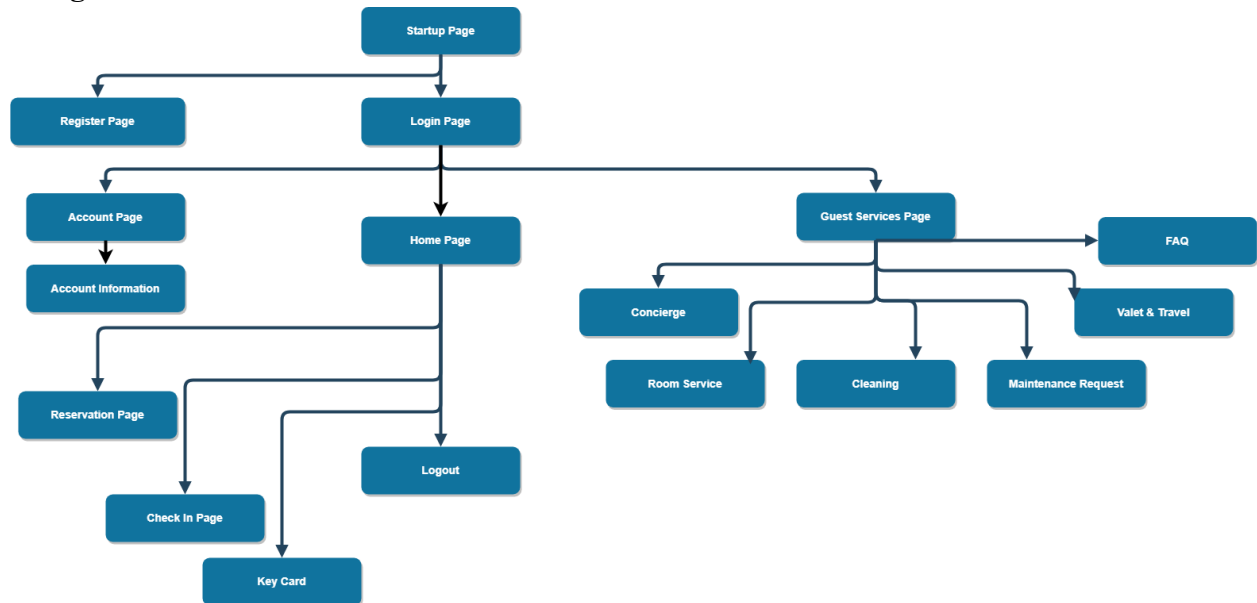
New In Progress Completed

Completed Requests:

Bottom navigation bar: User icon, Home icon, Menu icon

Completed Requests

## Use Effort Estimation Navigation Tree



Since we are building an android app navigation through each page is one click that will take you to a new page in the app. So from the guest services page to access each of the different features it is one button click. Each page also provides a back button to return to the previous one.

**\*\*Guest services page may have additional features in the future.**

## Project Management

	<b>Stakeholders / Actors</b>	<b>Use Cases</b>	<b>Use Case Diagram</b>	<b>Traceability Matrix</b>	<b>Screen Mockups</b>	<b>Use Effort Estimation</b>
<b>Keya</b>	20%	10%	10%	10%		10%
<b>Zain</b>	20%	10%	10%	10%		10%
<b>Mohammed</b>		10%	10%	10%		30%
<b>Purna</b>	20%	10%	10%	10%		10%
<b>Nga Man</b>		10%	10%	10%	30%	
<b>Rameen</b>		10%	10%	10%	30%	
<b>Shilp</b>	20%	10%	10%	10%	10%	
<b>Mathew</b>		10%	10%	10%	15%	15%
<b>Eric</b>		10%	10%	10%	15%	15%
<b>Thomas</b>	20%	10%	10%	10%		10%